

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of:

Tadashi OHASHI

Application No.:

Group Art Unit:

Filed: January 29, 2004

Examiner:

For: KNOWLEDGE PROCESSING SYSTEM

**SUBMISSION OF CERTIFIED COPY OF PRIOR FOREIGN
APPLICATION IN ACCORDANCE
WITH THE REQUIREMENTS OF 37 C.F.R. § 1.55**

Commissioner for Patents
PO Box 1450
Alexandria, VA 22313-1450

Sir:

In accordance with the provisions of 37 C.F.R. § 1.55, the applicant(s) submit(s) herewith a certified copy of the following foreign application:

Japanese Patent Application No(s). 2003-79466

Filed: March 24, 2003

It is respectfully requested that the applicant(s) be given the benefit of the foreign filing date(s) as evidenced by the certified papers attached hereto, in accordance with the requirements of 35 U.S.C. § 119.

Respectfully submitted,

STAAS & HALSEY LLP

Date:

1/30/04

By:

Richard A. Gollhofer

Richard A. Gollhofer
Registration No. 31,106

1201 New York Ave, N.W., Suite 700
Washington, D.C. 20005
Telephone: (202) 434-1500
Facsimile: (202) 434-1501

日 本 国 特 許 庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日 2 0 0 3 年 3 月 2 4 日
Date of Application:

出 願 番 号 特 願 2 0 0 3 - 0 7 9 4 6 6
Application Number:

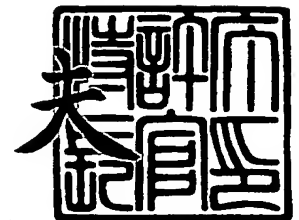
[ST. 10/C]: [J P 2 0 0 3 - 0 7 9 4 6 6]

出 願 人 富 士 通 株 式 会 社
Applicant(s):

2 0 0 3 年 1 2 月 9 日

特許庁長官
Commissioner,
Japan Patent Office

今 井 康



出証番号 出証特 2 0 0 3 - 3 1 0 1 6 4 5

【書類名】 特許願

【整理番号】 0253724

【提出日】 平成15年 3月24日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 17/30

【発明の名称】 知識処理システム

【請求項の数】 5

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

【氏名】 大橋 正

【特許出願人】

【識別番号】 000005223

【氏名又は名称】 富士通株式会社

【代理人】

【識別番号】 100074099

【住所又は居所】 東京都千代田区二番町8番地20 二番町ビル3F

【弁理士】

【氏名又は名称】 大菅 義之

【電話番号】 03-3238-0031

【選任した代理人】

【識別番号】 100067987

【住所又は居所】 神奈川県横浜市鶴見区北寺尾7-25-28-503

【弁理士】

【氏名又は名称】 久木元 彰

【電話番号】 045-573-3683

【手数料の表示】

【予納台帳番号】 012542

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9705047

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 知識処理システム

【特許請求の範囲】

【請求項 1】 設計対象に関する知識をクラス及びクラス間の関係によって構造的に構成する知識処理システムにおける知識処理方法であって、

該知識を記述するクラスを包括的に記述する名前を有したスーパークラスと、該クラスと、該クラス間の関係をデータベースに格納する格納ステップと、

該データベースに格納されているスーパークラスに関連するクラスを検出し、該クラス間の関係を多値論理に基づく推論によって生成し、該データベースに格納されている情報と、該推論によって得られた該クラス間の関係から該知識の構造を構築する推論ステップと

からなることを特徴とする方法をコンピュータに実行させるプログラム。

【請求項 2】 前記クラス間の関係には、前記多値論理に基づく推論によって得られたクラス間の関係及び階層構造を重み付ける重みが含まれることを特徴とする請求項 1 に記載のプログラム。

【請求項 3】 前記推論ステップでは、前記推論によってクラス間に関連づけ可能な新しいクラスを生成可能な場合には、該新しいクラスを生成し、他のクラスと関連付けることで、知識の構造化を行うことを特徴とする請求項 1 に記載のプログラム。

【請求項 4】 設計対象に関する知識をクラス及びクラス間の関係によって構造的に構成する知識処理システムにおける知識処理方法であって、

該知識を記述するクラスを包括的に記述する名前を有したスーパークラスと、該クラスと、該クラス間の関係をデータベースに格納する格納ステップと、

該データベースに格納されているスーパークラスに関連するクラスを検出し、該クラス間の関係を多値論理に基づく推論によって生成し、該データベースに格納されている情報と、該推論によって得られた該クラス間の関係から該知識の構造を構築する推論ステップと

からなることを特徴とする方法

【請求項 5】 設計対象に関する知識をクラス及びクラス間の関係によって構

造的に構成する知識処理システムであって、

該知識を記述するクラスを包括的に記述する名前を有したスーパクラスと、該クラスと、該クラス間の関係をデータベースに格納する格納手段と、

該データベースに格納されているスーパクラスに関連するクラスを検出し、該クラス間の関係を多値論理に基づく推論によって生成し、該データベースに格納されている情報と、該推論によって得られた該クラス間の関係から該知識の構造を構築する推論手段と

からなることを特徴とする知識処理システム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、オブジェクト指向言語を使った知識処理システムに関する。特に、マルチメディアオブジェクト指向言語を使った知識処理システムに関する。

【0002】

【従来の技術】

従来、設計者は業務開発やシステム開発等において筆記具を用いて手作業でブロック図を描画したり、または、パソコンやワークステーション等でUML (Unified Modeling Language) 等のオブジェクト指向のグラフィックエディタ機能を用いて描画していた。実世界の暗黙知（例えば、実世界の出来事を記述する人が持っている実世界に対する個人的な知識）や既成知（実世界の出来事を記述する人が実際に記載した知識）のモデリングのための知識及び知識間の相互関係や知識構造の階層関係やその知識及び知識間の関係及び知識構造の階層関係による知識を連ねたプロセス（流れ）やプログラム構造を前述の描画を利用して既成知として知識表現していた。この知識や知識構造を参照等する利用者は手作業で描画された図を目視的に自分の頭脳の中へ取り込んで頭脳の中で創造作業をなし、知識処理していた。利用者はまた、パソコンやワークステーション等を利用してその知識や階層構造からなる業務のプロセス（流れ）やプログラム構造を参照または利用し、更には必要に応じて修正を加えたりして利用者の頭脳の中に埋没されていた知識や勘や経験等の暗黙知を既成知として知識データベースへ取り込

んでいる。

【0003】

従来の技術を知識構造の設計側と利用側で分けてまとめると以下の通りとなる。

。

・設計側

従来のプログラム開発、業務開発などにUML等を用いて設計者がUMLエディタを用いて設計していた。

【0004】

その設計内容は以下の通りである。

(1) 構造的描写（ユースケース、クラス、インターフェース、コンポーネント、コラボレーション）を行っている。

(2) 振る舞いの描写（相互作用、状態マシーン）など

(3) グループ的描写（パッケージ）

(4) 注記的描写（ノート）

(5) 関係的描写（ユースケース、クラス、インターフェース、コンポーネント、コラボレーション）

(6) ダイアグラムの描写（ユースケース図、クラス図、オブジェクト図、シーケンス図、コラボレーション図、状態図、アクティビティ図、コンポーネント図、配置図）など

・利用側

以上の設計生成物を利用側のアプリケーションプログラムの利用や人間が設計生成物を目視することによって自らの手作業により利用していた。

【0005】

図13は、従来技術を説明する図である。

設計側のUML等による知識構造のモデリングは、各記述対象（event）に対して、クラスを生成し、これらのクラス間の関係を記述することによって行う。

【0006】

クラス間の関連の仕方としては、①依存、②汎化、③関連、④名前、⑤ロール、⑥多重度、⑦集約がある。

同図では、Eventというクラスに対応して、windowというクラスが依存関係を持っており、DialogBoxというクラスは、Controlというクラスと関連がある。CloseWindowsというクラスとDialogBoxというクラスは、大きく見ればwindowの制御にあたるので、一般に言うときwindowのクラスに含まれるべきものである。従って、DialogBoxとCloseWindowsは、汎化という関連づけにおいて、windowというクラスと関連付けられる。また、personというクラスは、Companyというクラスと関係があり、personがCompanyのために働くという関連づけが行われる。また、personやCompanyは、人の名前や会社の名前をインスタンスとして持つ。更に、personとCompanyには、ロール名が付けられる。この場合、personの方には、ロール名としてemployee、Companyの方にはemployerというロール名が付けられる。また、1つの会社に対する人の数に対応して、多重度が定められる。personに記述される人が1つの会社だけに属する場合には、personのCompanyへの多重度は1であるが、ある会社の従業員が100名であるとするとき、Companyへ関連付けられるpersonというクラスが100個存在し、Companyのpersonへの多重度は100となる。更に、会社は複数の部門からなるのが通常であるが、この場合、部門を記述するDepartmentというクラスは全て、Companyというクラスの基に集められて関連付けられるので、これを集約という。

【0007】

以上は、設計側が知識構造を設計する場合に使用するクラス間の関連づけの種類の例であるが、これらの関連づけを駆使することによって設計側では、様々な知識内容を設計することが出来る。一方、従来では、このようにして設計された知識内容を受け取った利用側では、この知識内容を読み取る場合、人間が自分の頭脳を使って読み解いていた。

【0008】

特許文献1においては、UMLなどのオブジェクト指向プログラムにおいて、マルチメディア設計情報に反映すべき情報を抽出するか否かの判断を自動的に行い、判断の精度を向上させる技術が記載されている。

【0009】

特許文献2においては、UMLなどを使ったオブジェクト指向設計の結果をテ

ストする場合において、単純な作業、参照項目の多い作業を支援し、テスト項目が不要に増大しないようにテスト項目を生成する技術が開示されている。

【0 0 1 0】

【特許文献 1】

特開 2 0 0 0 - 1 1 2 7 4 2 号公報

【特許文献 2】

特開 2 0 0 1 - 2 2 2 4 2 9 号公報

【0 0 1 1】

【発明が解決しようとする課題】

従来の UML などを使ったオブジェクト指向の知識設計における問題点を設計側と利用側に分けて示す。

・設計側

(1) UML は知識を構造的に記述可能な言語であるが、実世界における記述対象の多くは暗黙知であり、言語で必ずしも表現できない。

(2) UML は知識構造のビジュアル化のために使用可能な言語であるが、実世界においては、ビジュアル自身が確定的でなく抽象的である。

(3) UML は知識内容の仕様化のための言語であるが、仕様書の内容は全てが簡潔明瞭であり、論理的には不確定であってはならないのに対し、仕様化される知識内容である実世界は不確定な要素が多くある。

(4) UML は概念モデルを対象としているが、概念自身は当事者の主観によることが多いため、第三者によってゆがめられやすいものとなっている。

【0 0 1 2】

設計に関しては、既存のツールでは、どのように設計するかなど全てが利用側にまかされているので、設計者の負担が大きい。

・利用側

設計側で設計された知識内容の利用に関しては既存のツールでは全て利用側にまかされているので、利用者の負担が大きい。

【0 0 1 3】

本発明の課題は、設計者、利用者の負担を減らしつつ、正確に知識設計するこ

との出来る知識処理システムを提供することである。特に、その知識処理システムは、マルチメディア知識処理システムである。

【 0 0 1 4 】

【課題を解決するための手段】

本発明の知識処理システムは、設計対象に関する知識をクラス及びクラス間の関係によって構造的に構成する知識処理システムであって、該知識を記述するクラスを包括的に記述する名前を有したスーパークラスと、該クラスと、該クラス間の関係をデータベースに格納する格納手段と、該データベースに格納されているスーパークラスに関連するクラスを検出し、該クラス間の関係を多値論理に基づく推論によって生成し、該データベースに格納されている情報と、該推論によって得られた該クラス間の関係から該知識の構造を構築する推論手段とからなることを特徴とする。

【 0 0 1 5 】

本発明によれば、スーパークラスという知識構造を代表するクラスを設け、スーパークラスに関連するか否かという形でクラスを関連付けることによって知識構造を記述するクラスを構造化する。更に、クラスの構造化において、設計者が設計したクラス間の関係の他に、多値論理を用いた、機械による推論の結果得られた新しいクラス間の関係も当該知識構造の一部とする。

【 0 0 1 6 】

更に他の側面によれば、多値論理を用いた推論によって、新たなクラスを生成し、知識構造をより精密化し、暗黙知の部分を減らし、できるだけ設計者の主観によらない既成知で知識を構造化することによって、当該知識構造によって記述される知識内容を利用者によりわかりやすく、正しく伝えることが出来る。この場合、利用側でも設計側で生成された知識構造に加え、利用側で推論した結果を知識構造に加えることも可能である。

【 0 0 1 7 】

【発明の実施の形態】

本発明では、知識データベースにある、またはW e b 空間にあるコードネーム等のネーミングを持つ（例えば、ある開発プロジェクトにおいて、知識構造が設

計される場合、その開発プロジェクトに一意に付与された開発コードネームのネーミングを持つ)、包括的な、知識の上位概念に対応するクラス(スーパークラス)を、従来の技術で示した知識構造化のための機能を持つ知識として扱い、設計者の設計段階及び利用者の利用段階時に重み付けがなされた知識間の相互関係、重み付けされた階層関係、等を時制推論で非単調的に自動生成させ、その結果得られた新たな知識を相互関係または階層関係に自動挿入させる。

【0018】

本発明の実施形態に従った設計側と利用側の基本構成を以下に示す。

・設計側

- (1) コードネームをスーパークラスにセットする。
- (2) 重み付きUMLを編集する機能を持つ。
- (3) 重み関係(3要素)処理機能を持つ(依存、汎化、関連)。
- (4) 変数の多値論理を推論する機能を持つ(ファジーロジックやジェネティックアルゴリズム)。
- (5) 時制推論機能を持つ。
- (6) RDFスキーマ(Resource Description Framework schema)あるいはXMLスキーマなどに出力するWebコンパイラ機能を持つ。

・利用側

- (1) コードネームをスーパークラスとする。
- (2) UMLを編集する機能を持つ。
- (3) 重み関係(3要素)処理機能を持つ(依存、汎化、関連)。
- (4) 変数の多値論理を推論する機能を持つ(ファジーロジックあるいはジェネティックアルゴリズム)。

【0019】

以上の構成を設けることにより、以下のような作用が得られる。

・設計側

- (1) スーパークラスを頂点とする知識構造で全ての概念を表現することができる。
- (2) 設計された知識構造を表記したUMLを編集することができる。

- (3) 知識構造の関係 (3 要素) を重み付けを用いて処理することができる。
- (4) 重み付けを多値論理 (ファジーロジック等) で推論することができる。
- (5) シーケンス図の処理過程でも上記多値論理を時制推論することができる。
- (6) 推論した結果により新たにクラスを生成することができる。
- (7) 以上の結果を R D F スキーマあるいは X M L スキーマに依拠して W e b コンパイルすることができる。

・ 利用側

- (1) スーパクラスを頂点とする知識構造で知識表現としてとらえることができる。
- (2) 幾つかのクラスを基に新たなクラスを推論することができる。

【 0 0 2 0 】

図 1 は、本発明の実施形態の要部を説明する図である。

図 1 において、window というスーパクラスが存在するとする。このスーパクラスを基に、多値論理を用いて推論することにより、他のクラスとの関係を推論し、関連付ける。図 1 においては、CloseWindows というクラスは、多値推論により、スーパクラスである window に対して、汎化という関係で関連付けられる。また、DialogBox というクラスもスーパクラスである window に汎化という関係で関連付けられる。更に、スーパクラス window は、推論により、Event というクラスと関連付けられ、DialogBox というクラスは、Control というクラスと関連付けられる。また、CloseWindows と DialogBox の関係にも、多値論理による推論が行われるが、図 1 の場合、関連が発見されなかった場合となっている。

【 0 0 2 1 】

また、多値論理による関連づけがなされた場合には、関連の重みが計算され、データとしてクラスの関連データに付け加えられる。更に、多値論理による推論の結果、設計者は作成していないが、クラス間の関連の中で、新しいクラスが生成される場合がある。例えば、「手帳」というクラスと「テープレコーダ」というクラスが設計者によって、「ミーティング」というクラスの下に関連付けられて生成されていた場合、多値論理による推論により、「手帳」と「テープレコーダ」の共通の特徴である「記録」という概念に対応し、新しいクラスが「手帳」

と「テーブルコード」の汎化の結果として生成される。これは、「手帳」というクラスのインスタンスと「テーブルコード」というクラスのインスタンスに共通に「記録」というインスタンスが有る場合、この「記録」という名前のクラスを生成することによって、「手帳」と「テーブルコード」を「ミーティング」とは異なる視点の分類である「記録」という上位概念によって構造化する事が出来る。このような、複数の視点からの構造化を自動生成することにより、設計者の知識構造化作業の負担を減らすことができる。

【0022】

図1の利用側では、設計側で作成された知識構造を基に、やはり、多値論理による推論を使って、可能な場合には新しいクラスを生成する。利用側では、設計側で作った知識構造をそのまま使うこともできるが、新たなクラスを生成することにより、構造化がより整理され、設計側の作った知識構造を理解するのに役立つ。また、時制推論を使用することにより、例えば、製品開発の段階を記述した知識構造の場合、アイデアの創出段階では、スケッチが入手可能であるが、モデルや製品は存在しない、アイデアの具体化段階では、スケッチとモデルが入手可能であるが、製品は存在しない、製造段階では、スケッチ、モデル、設計図が入手可能であるが、製品は存在しない、開発終了段階では、スケッチ、モデル、設計図、製品の全てが入手可能であるなど、時間の経過と共に、どのアイテムが入手可能ななどの推論を行う。時制推論は、「アイデア創出段階」、「アイデア具体化段階」、「製造段階」、「開発終了段階」などのクラスを発見し、クラスの有する情報から、これらの時制を推論することにより行う。

【0023】

図2～図8は、本発明の実施形態を具体的に適用した例を説明する図である。

この適用例では、装置などの開発時に付与される開発コードネーム（DCN：Development Code Name）をスーパークラスに設定する場合について述べる。

【0024】

DCNは設計側、利用者側がクラス／オブジェクトの名前、属性（property）、登録者職制をWebで知識データベースへ登録するものであり、同一名の重複回避または重複時のドメイン設定を行うシステムで構成されている。開発コード

ネームはスーパークラスに位置し、その配下に構成要素のクラスまたはサブクラスをクラスタとして連ねる。構成要素のつながりが機能的振る舞いとなる際の出力は実際のWebの利用者へと渡す。例えば、利用者は自ら主観（暗黙知）に照らし合わせた、知識の獲得が可能となる。

【0025】

図2及び図3は、全体の流れを示すブロック図である。

設計側においては、設計業務において、開発コードネームの設定機能10、クラス名やオブジェクト名を格納する名前テーブル11、UMLエディタなどの画面読み取り機構12などを備えた端末を使う。これらを用いて、シーケンス図や業務プロセスを示す図を生成し、これらのクラス名やオブジェクト名などを名前知識データベース13に格納する。また、画面読み取り機構12からは、時間／処理知識データベース25に、各クラスの時制や処理内容を格納する。

【0026】

名前知識データベース13から、シーケンシャル名前呼び出し機構14がクラス名やオブジェクト名をシーケンシャルに呼び出し、相互関係組み合わせテーブル作成15を行う。相互関係組み合わせテーブルは、組み合わせテーブル16に格納される。

【0027】

結合推論機構18では、時間／処理知識データベース25、組み合わせテーブル16、及び命題ルール知識データベース17から、それぞれ各クラスの時制、処理内容データ、クラス間の相互関係データ、命題ルールを読み込み、ファジー推論やジェネティックアルゴリズムによる推論などの多値論理による推論を行う。この結果、設計されたクラスやオブジェクトが知識階層構造として構成される他、クラス間やオブジェクト間の関連の重みが計算される。知識階層構造となったクラスやオブジェクトは、クラス／オブジェクト知識データベース19に格納され、重みは重み知識データベース22に格納される。

【0028】

時間／処理知識データベース25、及び重み知識データベース22のデータ、クラス／オブジェクト知識データベース19のマルチメディアのオブジェクトを

組み込み20が行われたデータは、Webコンパイラ24によって、Web表示用にコンパイルされ、インターネットなどのネットワークを介して、利用側に出力成果物として提供される。もちろん、出力成果物はネットワーク上に適用されなくても良い。その場合には、DVDなどの記録媒体に出力成果物を記録して、利用側に出力成果物を渡す。

【0029】

利用側では、設計側に対応する、時間／処理知識データベース25' 名前データ13'、重みデータ22'、及びクラス／オブジェクト知識データベース19' が設けられ、設計側からのデータを入力成果物として格納する。時間／処理知識データベース25' のデータと名前知識データベース13' のデータとでシーケンス図や業務プロセスを示す図などが構成される。知識階層構造を有するクラス／オブジェクトは、クラス／オブジェクト知識データベース19' に格納されており、開発コードネームや名前検索によって、クラス／オブジェクト検索30を行えるようになっている。

【0030】

時間／処理データ25'、重み知識データベース22' 及びクラス／オブジェクト知識データベース19' のデータは、結合推論機構27において、検索されたスーパークラスの多値論理による推論が行われる。この場合、結合推論機構27においては、ユーザ／アプリケーション処理ルール知識データベース26とのやりとりで、クラス間関係の検索28を行うこともできる。結合推論機構27は、ファジーロジックやニューラルネットワーク、ジェネティックアルゴリズムなどを用いた多値論理の推論機構であり、クラス間、オブジェクト間などの関係を推論する。これにより、設計側から得たデータからクラス間あるいはオブジェクト間の関係の再構築29を行い、重みを重み知識データベース22' に、知識階層構造となったクラスまたはオブジェクトをクラス／オブジェクト知識データベース19' に格納する。利用者は、開発コードネームや名前検索で検索したクラスやオブジェクト、スーパークラスに関連した再構築された知識構造をマルチメディアによってオブジェクト表示31させる。これにより、設計側で設計された構造化された知識情報が利用者側に表示される。このマルチメディア表示には、画像

による表示の他、音響やテキストなどによる知識の提示も含む。

【0031】

なお、本発明の実施形態において使用する推論においては、多値論理を使っているため非単調推論ができる。つまり二つの命題は人間が読み書きした場合明らかに偽であると分かっても、命題論理にして処理すると二つの命題が共に真となることがある。しかし、多値論理を使用すると重み出力があるので、二つの命題が真であっても重みの大きい方を真ととらえる（非単調）ことができる。この結果、信頼度が高く精度の高い推論が可能となる。

【0032】

図4は、本発明の実施形態に従った設計側の処理フローである。

まず、入力として、開発コードネームや名前リストを設計側の端末に入力する。ステップS10では、この入力を受けて、クラス群をスキャンし、スーパークラスを見つける。ステップS11においては、スーパークラスに関係した隣接クラスを見つける。そして、ステップS12において、クラス間の結合関係を得、ステップS13において、ファジーロジックやニューラルネットワーク、ジェネティックアルゴリズムなどを使った推論を行う。ステップS14において、推論の結果をクラス間の関係の結果とする。この時、推論にファジーロジックを使った場合には、各関係に重みが付く。

【0033】

そして、ステップS15において、新しいクラスを生成すべきか否かが判断され、ステップS15の判断がYESの場合には、ステップS16において、新クラスを生成し、ステップS17に進む。ステップS15の判断がNOの場合には、そのままステップS17に進む。ステップS17においては、全てのクラスについて処理が終わったか否かを判断する。ステップS17における判断がNOの場合には、ステップS13に進む。ステップS17における判断がYESの場合には、ステップS18に進む。ステップS18においては、全結合について処理し終わったか否かが判断される。ステップS18の判断がNOの場合には、ステップS11に戻る。ステップS18における判断がYESの場合には、処理を終了する。

【0034】

図5は、開発コードネーム知識データベースの例を示した図である。

このデータベースにおいては、開発コードネームDCNがクラス／オブジェクト名として格納されると共に、開発コードネームに対応して、プロパティが格納される。プロパティにおいては、登録職制、部署名、装置名、型番、機能概略、構成、適用、ユーザ、環境が登録される。

【0035】

図6は、本発明の実施形態に従った利用側の処理フローである。

利用側では、開発コードネームや名前リストを利用側の端末に入力する。ステップS20においては、クラス群のスキャンを行い、スーパークラスを見つける。ステップS21においては、スーパークラスの隣接クラスを見つける。ステップS22においては、クラス間の結合関係を取得する。ステップS23においては、命題ルール端末に与え、ステップS24においては、命題ルールに合致するクラスを得る。命題ルールは、「a」ならば「b」、かつ、「b」ならば「c」、故に「a」ならば「c」というような命題を記述するルールである。これによれば、図6左に記載されているクラス間の関係において、あるクラス「a」がクラス1と関係有り、クラス1がクラス2と間接的な関係にあり、クラス「b」がクラス2と関係があり、クラス2と関連するクラス3がクラス4と間接的な関係にあり、クラス4とクラス「c」が関係があるならば、クラス「a」とクラス「c」は間接的な関係があるということになる。ステップS25において、クラスとクラスの関係性を推論する。この推論は、ファジーロジックやニューラルネットワークやジェネティックアルゴリズムで行う。ステップS26においては、新しいクラスを生成可能か否かが判断され、YESの場合には、ステップS27で新クラスを生成してステップS28に進み、NOの場合には、そのまま、ステップS28に進む。

【0036】

ステップS28においては、全てのクラスについて処理が終わったか否かが判断され、NOの場合にはステップS23に進み、YESの場合にはステップS29に進む。ステップS29においては、全結合の処理が終わったか否かが判断さ

れる。ステップ S 2 9 の判断が N O の場合には、ステップ S 2 1 に戻り、Y E S の場合には、処理を終了する。

【0037】

図 7 は、時制推論を説明する図である。

同じ推論ルールであっても、業務プロセスなどがどの段階にあるかによって結果が異なる。ある時点 t 1 では、「a」ならば「b」、且つ、「b」ならば「c」であれば、「a」は「c」が成立する場合でも、ある時点 t 2 では、「a」ならば「b」、且つ、「b」ならば「c」であっても「a」は「c」とはならない。

【0038】

このように、時制を考慮して推論ルールを構築しておけば、時制を含んだ推論を行うことが出来る。

図 8 は、設計側で得られた知識構造を利用者側に W e b データとして渡す場合を説明する図である。

【0039】

設計側では、設計側で知識構造をクラスとクラス間の関係（依存、汎化、関連、名前、ロール、多重度、集約）を入力し、多値論理による推論によって知識階層構造を生成する。設計側では、これを R D F スキーマあるいは X M L スキーマを使って、W e b コンパイルすると共に、U M L エディタを使って、クラス間関係から時制論理推論を行い、W e b コンパイルしてネットワークにデータを公開する。利用側では、これをネットワーク経由で受け取って、時制推論や脱多値論理化処理を行い、業務プロセスなどの構造化知識を端末に表示させる。

【0040】

図 9 ～図 1 2 は、知識構造とクラスの間を説明する図である。

図 9 に示されるように、「P C B」というインスタンスを共通に持つクラスが複数有る場合には、推論メカニズムにより、新しいクラス「P C B」が生成される。特に、本発明の実施形態では、ファジーロジックを使った場合、新たに生成されるクラス「P C B」への、元のクラスからの各関連データには重みが付加される（図 1 0 参照）。また、図 1 1 においては、Server というクラスと Locker と

いうクラスが関連付けられている場合、推論により、Serverの部分クラスであるLockerからPowSupというクラスへの関係がある重みで生成され、更に、推論により、PowSupというクラスとServerというクラスが関連付けられる。この場合、PowSupは、Serverの部分クラスであるとされる。

【 0 0 4 1 】

また、図 1 2 に示されるように、Serverの部分クラスである女性用ロッカーの知識を記述するクラスLockerには、色、高さ、奥行き、長さなどのプロパティが設けられ、これらは、Enumerationというクラスによって登録されるインスタンスの値が与えられる。Heightには、更に、他のEnumerationクラスが関連付けられ、背の高さの情報が細かく規定される。

【 0 0 4 2 】

このように、多値推論処理を行うことにより、設計側の暗黙知であり、設計側で関連付けられていなかったクラスが、関連付けられたり、新たなクラスが生成されるので、設計側の暗黙知をシステムが既成知としてクラスに記述することができる。既成知となった暗黙知は、利用側で紛れなく認識されるので、設計側で生成された知識構造がゆがめられることなく、利用側に渡される。

【 0 0 4 3 】

(付記 1) 設計対象に関する知識をクラス及びクラス間の関係によって構造的に構成する知識処理システムにおける知識処理方法であって、

該知識を記述するクラスを包括的に記述する名前を有したスーパクラスと、該クラスと、該クラス間の関係をデータベースに格納する格納ステップと、

該データベースに格納されているスーパクラスに関連するクラスを検出し、該クラス間の関係を多値論理に基づく推論によって生成し、該データベースに格納されている情報と、該推論によって得られた該クラス間の関係から知識構造を構築する推論ステップと

からなることを特徴とする方法をコンピュータに実行させるプログラム。

【 0 0 4 4 】

(付記 2) 前記クラス間の関係には、前記多値論理に基づく推論によって得られたクラス間の関係及び階層構造を重み付ける重みが含まれることを特徴とす

る付記 1 に記載のプログラム。

【0045】

(付記 3) 前記推論ステップでは、前記推論によってクラス間に関連づけ可能な新しいクラスを生成可能な場合には、該新しいクラスを生成し、他のクラスと関連付けることで、知識の構造化を行うことを特徴とする付記 1 に記載のプログラム。

【0046】

(付記 4) 前記推論ステップでは、クラス群で記述される知識構造について、時制推論を行い、記述されている知識の時間経過を追った変化を考慮したクラス間の関係を知識構造に組み込むことを特徴とする付記 1 に記載のプログラム。

【0047】

(付記 5) 設計側で生成された前記知識構造を利用する利用側において、該知識構造を記述するスーパークラス及びクラス群の情報から、多値論理を用いた推論により該設計側で設計された知識構造を再構築することを特徴とする付記 1 に記載のプログラム。

【0048】

(付記 6) 設計対象に関する知識をクラス及びクラス間の関係によって構造的に構成する知識処理システムにおける知識処理方法であって、

該知識を記述するクラスを包括的に記述する名前を有したスーパークラスと、該クラスと、該クラス間の関係をデータベースに格納する格納ステップと、

該データベースに格納されているスーパークラスに関連するクラスを検出し、該クラス間の関係を多値論理に基づく推論によって生成し、該データベースに格納されている情報と、該推論によって得られた該クラス間の関係から該知識の構造を構築する推論ステップと

からなることを特徴とする方法

(付記 7) 前記クラス間の関係には、前記多値論理に基づく推論によって得られたクラス間の関係及び階層構造を重み付ける重みが含まれることを特徴とする付記 6 に記載の方法。

【0049】

(付記 8) 前記推論ステップでは、前記推論によってクラス間に関連づけ可能な新しいクラスを生成可能な場合には、該新しいクラスを生成し、他のクラスと関連付けることで、知識の構造化を行うことを特徴とする付記 6 に記載の方法。

【0050】

(付記 9) 前記推論ステップでは、クラス群で記述される知識構造について、時制推論を行い、記述されている知識の時間経過を追った変化を考慮したクラス間の関係を知識構造に組み込むことを特徴とする付記 6 に記載の方法。

【0051】

(付記 10) 設計側で生成された前記知識構造を利用する利用側において、該知識構造を記述するスーパークラス及びクラス群の情報から、多値論理を用いた推論により該設計側で設計された知識構造を再構築することを特徴とする付記 6 に記載の方法。

【0052】

(付記 11) 設計対象に関する知識をクラス及びクラス間の関係によって構造的に構成する知識処理システムであって、

該知識を記述するクラスを包括的に記述する名前を有したスーパークラスと、該クラスと、該クラス間の関係をデータベースに格納する格納手段と、

該データベースに格納されているスーパークラスに関連するクラスを検出し、該クラス間の関係を多値論理に基づく推論によって生成し、該データベースに格納されている情報と、該推論によって得られた該クラス間の関係から該知識の構造を構築する推論手段と

からなることを特徴とする知識処理システム。

【0053】

(付記 12) 前記クラス間の関係には、前記多値論理に基づく推論によって得られたクラス間の関係及び階層構造を重み付ける重みが含まれることを特徴とする付記 11 に記載の知識処理システム。

【0054】

(付記 13) 前記推論手段では、前記推論によってクラス間に関連づけ可能

な新しいクラスを生成可能な場合には、該新しいクラスを生成し、他のクラスと関連付けることで、知識の構造化を行うことを特徴とする付記 1 1 に記載の知識処理システム。

【 0 0 5 5 】

(付記 1 4) 前記推論手段では、クラス群で記述される知識構造について、時制推論を行い、記述されている知識の時間経過を追った変化を考慮したクラス間の関係を知識構造に組み込むことを特徴とする付記 1 1 に記載の知識処理システム。

【 0 0 5 6 】

(付記 1 5) 設計側で生成された前記知識構造を利用する利用側において、該知識構造を記述するスーパークラス及びクラス群の情報から、多値論理を用いた推論により該設計側で設計された知識構造を再構築することを特徴とする付記 1 1 に記載の知識処理システム。

【 0 0 5 7 】

【発明の効果】

- (1) 知識の獲得に利用する UML に暗黙知が取り込んで表現できる。
- (2) 重み付け及び多値論理を UML を用いてビジュアル化された言語で利用できる所以、確定的な抽象的知識であっても取りこぼすことが無くなる。
- (3) UML で仕様化させるときの言語として利用できる所以、仕様書の内容は簡潔明瞭であり、論理的には不確定であっても良いので、実世界の不確定な要素に対しても知識の獲得対象になる。
- (4) UML は概念モデルを対象としてきたので、概念自身は当事者の主観によるところが多くても、第三者によって知識がより精緻化することが可能となる。
- (5) 利用側の負担が軽減される。
- (6) 従来は手作業で関係付けを行っていたが設計側の習熟度により本質が落ちこぼれることなくエキスパートが作業したような幅の広い的確なクラス及びオブジェクトの入手が可能となる。例えば、製品の障害が発生して故障原因を探索する場合のドキュメントや画像や音声のガイドブックを入手する場合に的確なオブジェクトを入手可能となる。

【図面の簡単な説明】**【図 1】**

本発明の実施形態の要部を説明する図である。

【図 2】

本発明の実施形態を具体的に適用した例を説明する図（その 1）である。

【図 3】

本発明の実施形態を具体的に適用した例を説明する図（その 2）である。

【図 4】

本発明の実施形態を具体的に適用した例を説明する図（その 3）である。

【図 5】

本発明の実施形態を具体的に適用した例を説明する図（その 4）である。

【図 6】

本発明の実施形態を具体的に適用した例を説明する図（その 5）である。

【図 7】

本発明の実施形態を具体的に適用した例を説明する図（その 6）である。

【図 8】

本発明の実施形態を具体的に適用した例を説明する図（その 7）である。

【図 9】

知識構造とクラスの間係を説明する図（その 1）である。

【図 1 0】

知識構造とクラスの間係を説明する図（その 2）である。

【図 1 1】

知識構造とクラスの間係を説明する図（その 3）である。

【図 1 2】

知識構造とクラスの間係を説明する図（その 4）である。

【図 1 3】

従来技術を説明する図である。

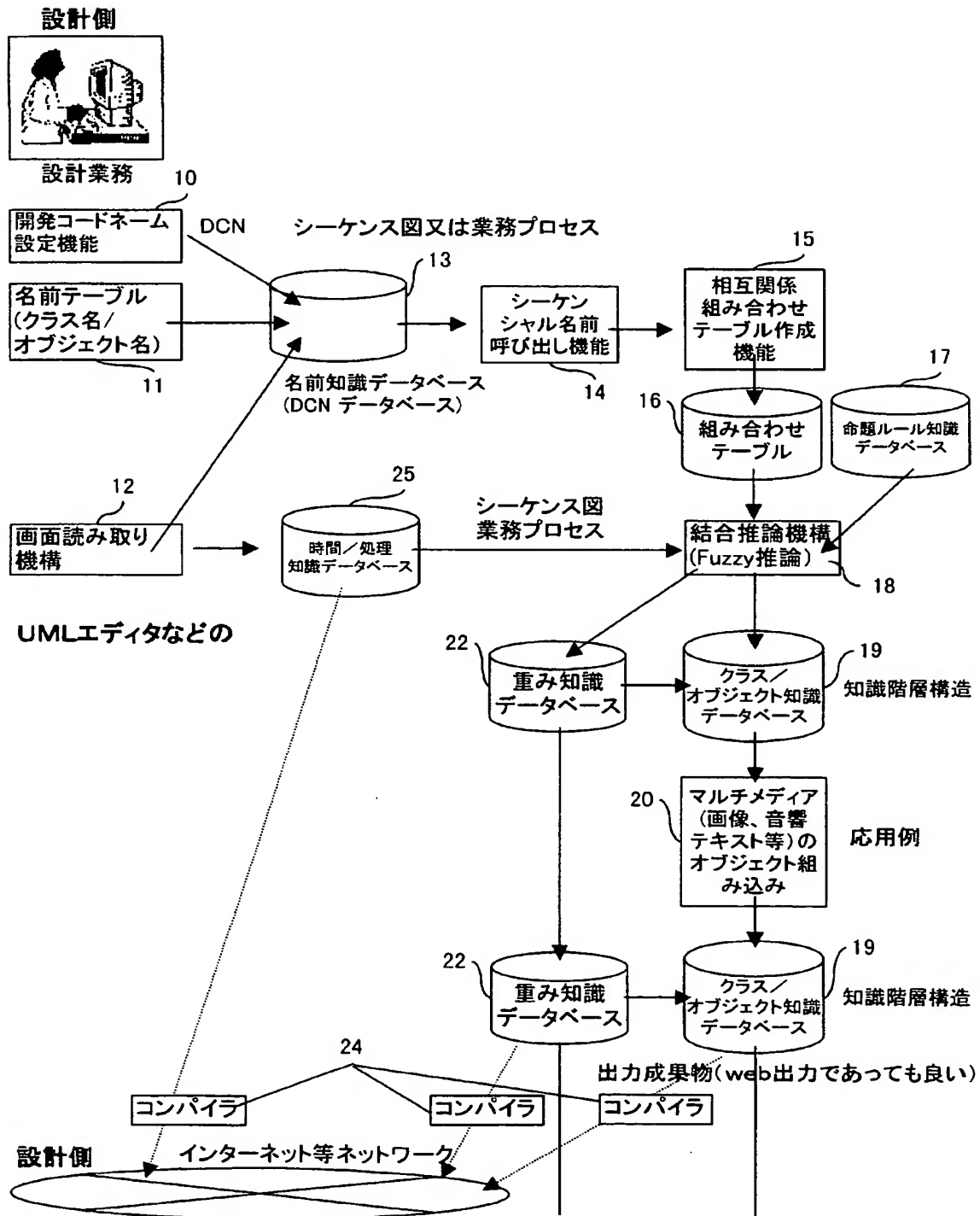
【符号の説明】

1 0 開発コードネーム設定機能

- 1 1、 名前テーブル
- 1 2 画面読み取り機構
- 1 3、 1 3' 名前知識データベース
- 1 4 シーケンシャル名前呼び出し機能
- 1 5 相互関係組み合わせテーブル作成
- 1 6 組み合わせテーブル
- 1 7 命題ルール知識データベース
- 1 8 結合推論機構
- 1 9、 1 9' クラス／オブジェクト知識データベース
- 2 0 マルチメディアのオブジェクト組み込み
- 2 2、 2 2' 重み知識データベース
- 2 4 W e b コンパイラ
- 2 5、 2 5' 時間／処理知識データベース
- 2 6 ユーザ／アプリケーション処理ルール知識データベース
- 2 7 結合推論機構
- 2 8 クラス間関係の探索
- 2 9 関係の再構築
- 3 0 クラス／オブジェクト検索
- 3 1 マルチメディアのオブジェクト表示

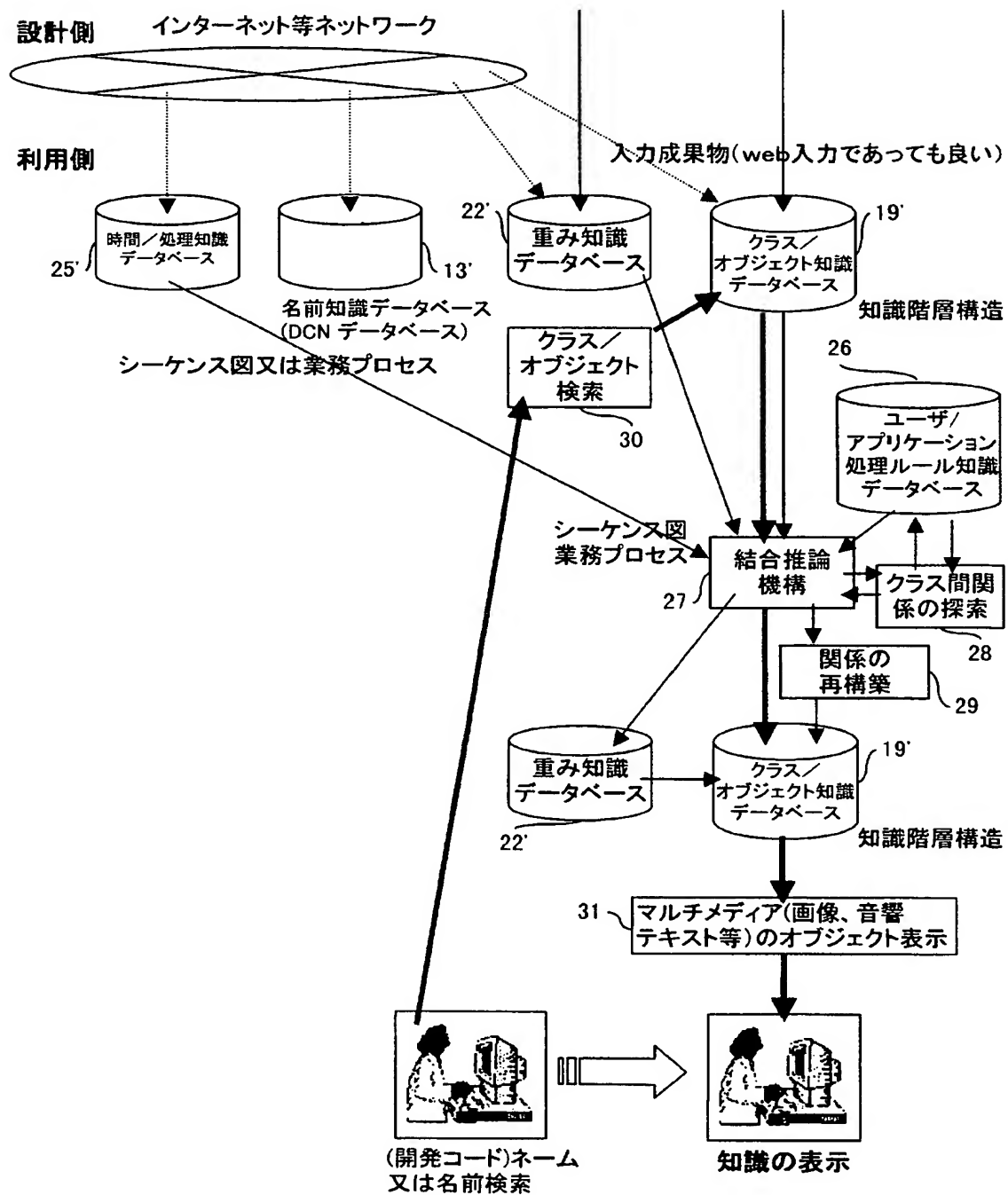
【図 2】

本発明の実施形態を具体的に適用した 例を説明する図(その1)



【図 3】

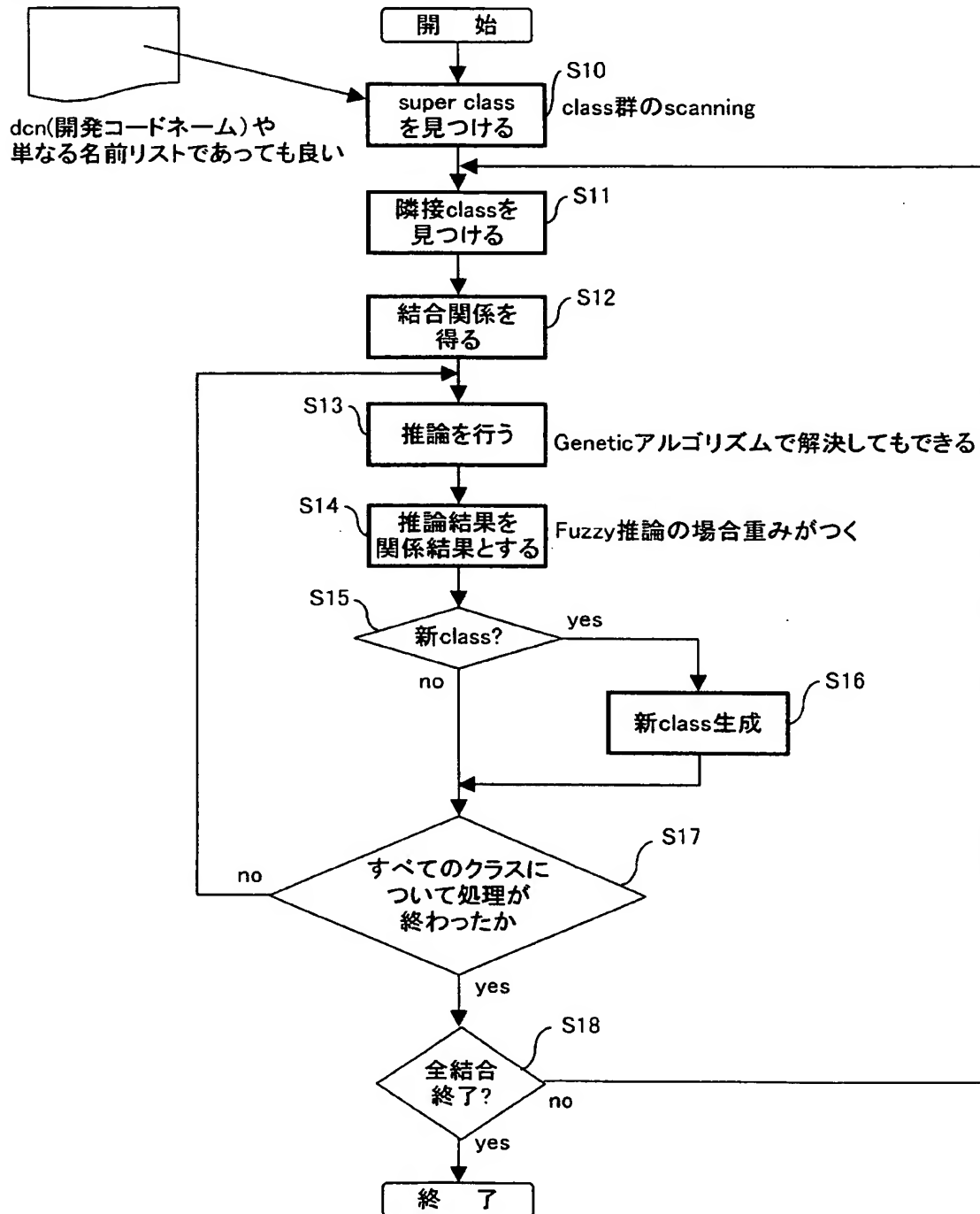
本発明の実施形態を具体的に適用した
例を説明する図(その2)



【図 4】

本発明の実施形態を具体的に適用した
例を説明する図(その3)

設計側



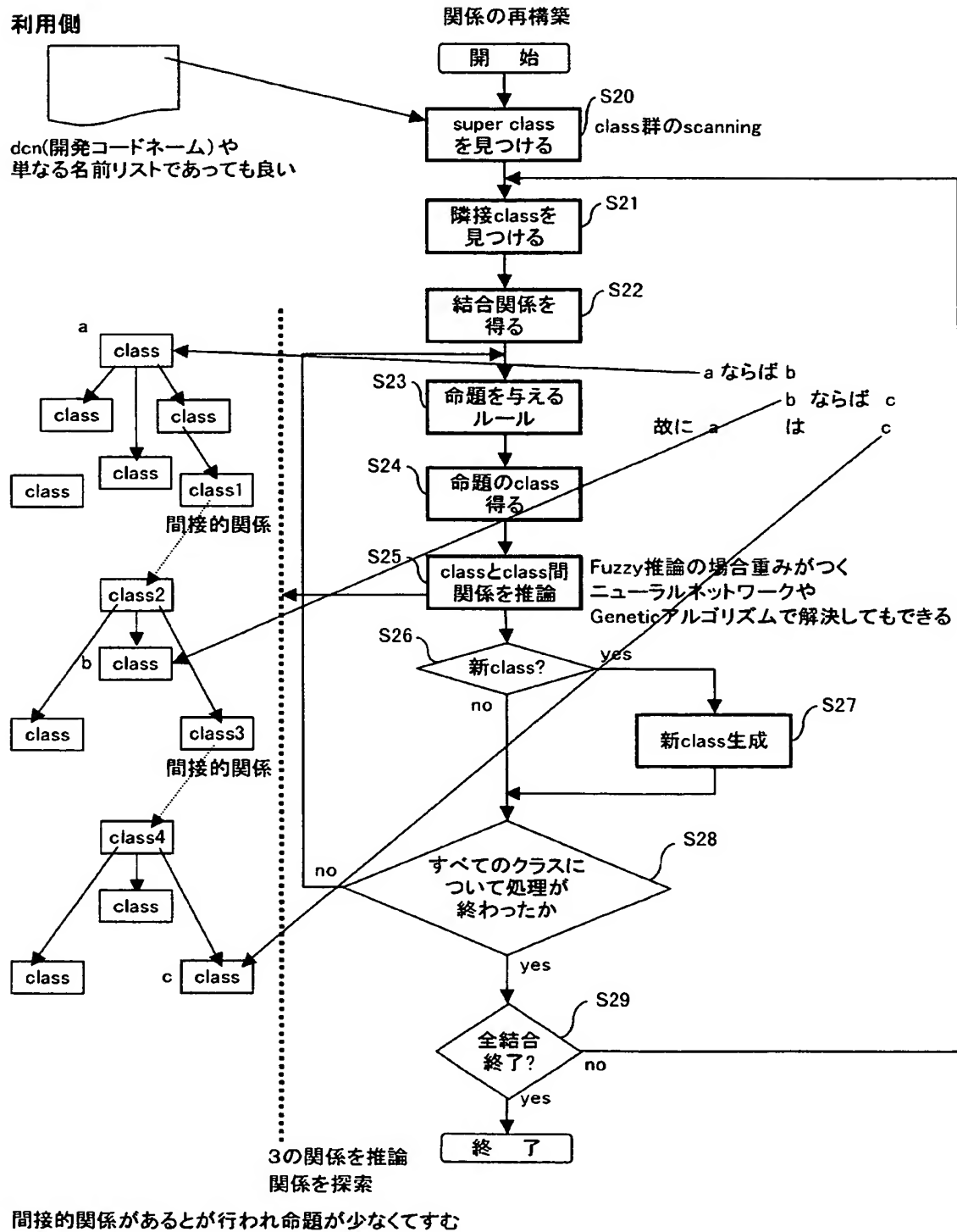
【図 5】

本発明の実施形態を具体的に適用した
例を説明する図(その4)

DCN (class/obj名)	property etc.	
Eagle	登録職制	VPS)XXX
	部長名	△△△△
	装置名	未定
	型番	未定
	機能概略	電子機器の 冷却装置
	構成	空冷Fan を使用
	適用	大型サーバ へ適用
Tomato	ユーザ	社内外用
	環境	XX規格に 準拠
Tomato	登録職制	VPS)XXX
	部長名	△△△△
	装置名	

【図 6】

本発明の実施形態を具体的に適用した 例を説明する図(その5)

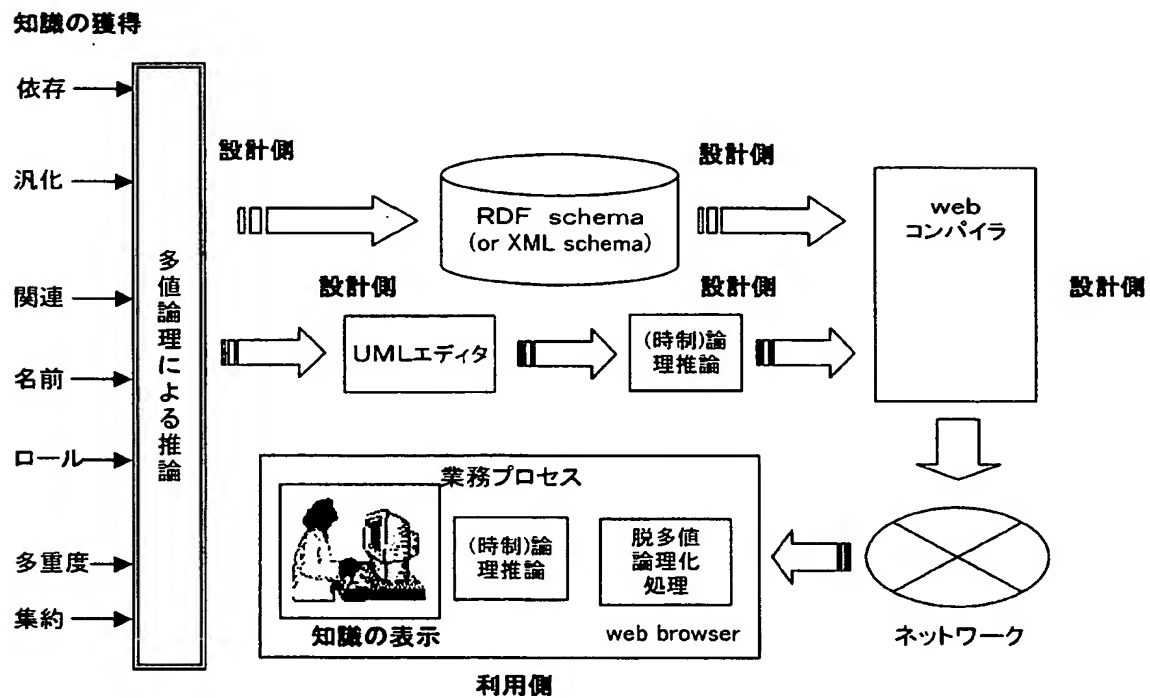


【図 7】

本発明の実施形態を具体的に適用した
例を説明する図(その6)

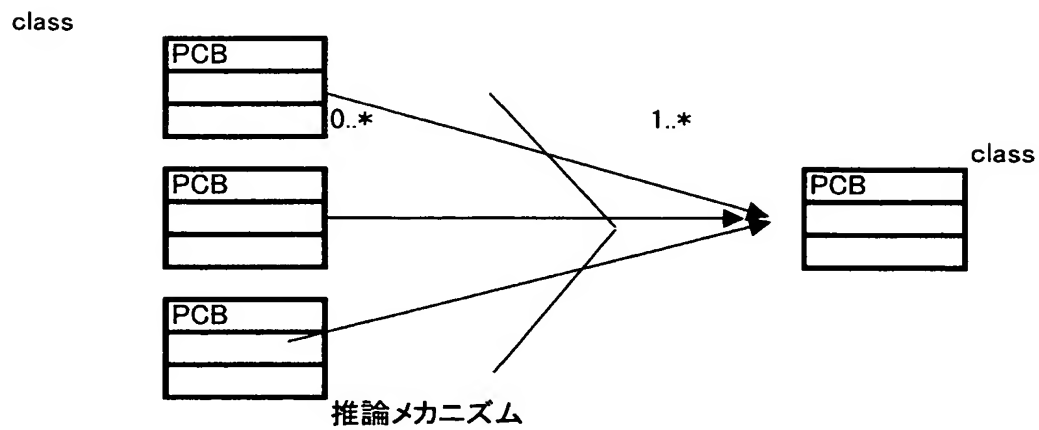
時間	t1	t2	t3	t4	t5
t1時点で		a ならば	b		
t1時点で			b ならば	c	
t1時点で		a	は	c	成立するが
		t2時点で	a ならば	b	
		t2時点で		b ならば	c
		t2時点では	a	は	c 成立しない

【図 8】

本発明の実施形態を具体的に適用した
例を説明する図(その7)

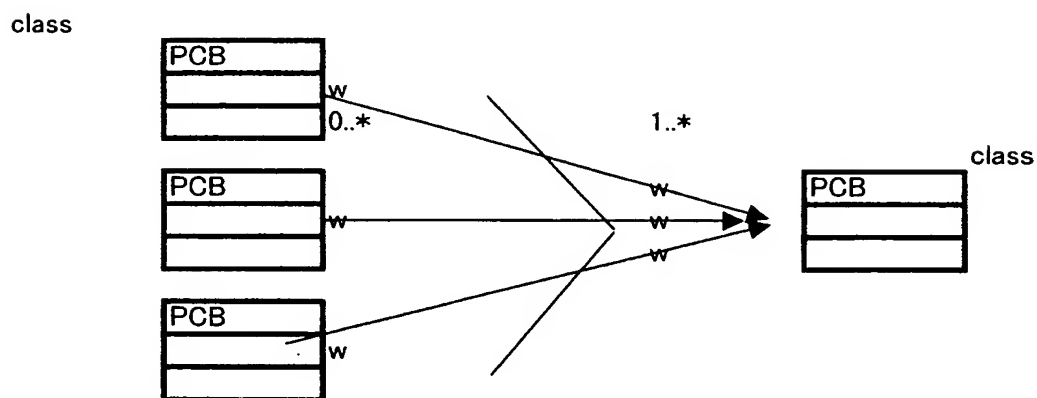
【図 9】

知識構造とクラスの関係を示明する図(その1)



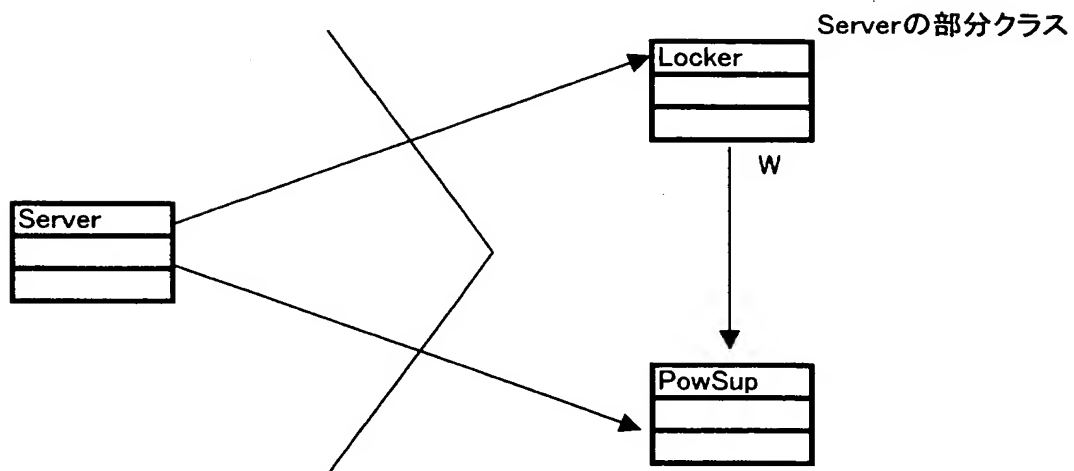
【図 10】

知識構造とクラスの関係を示す図(その2)



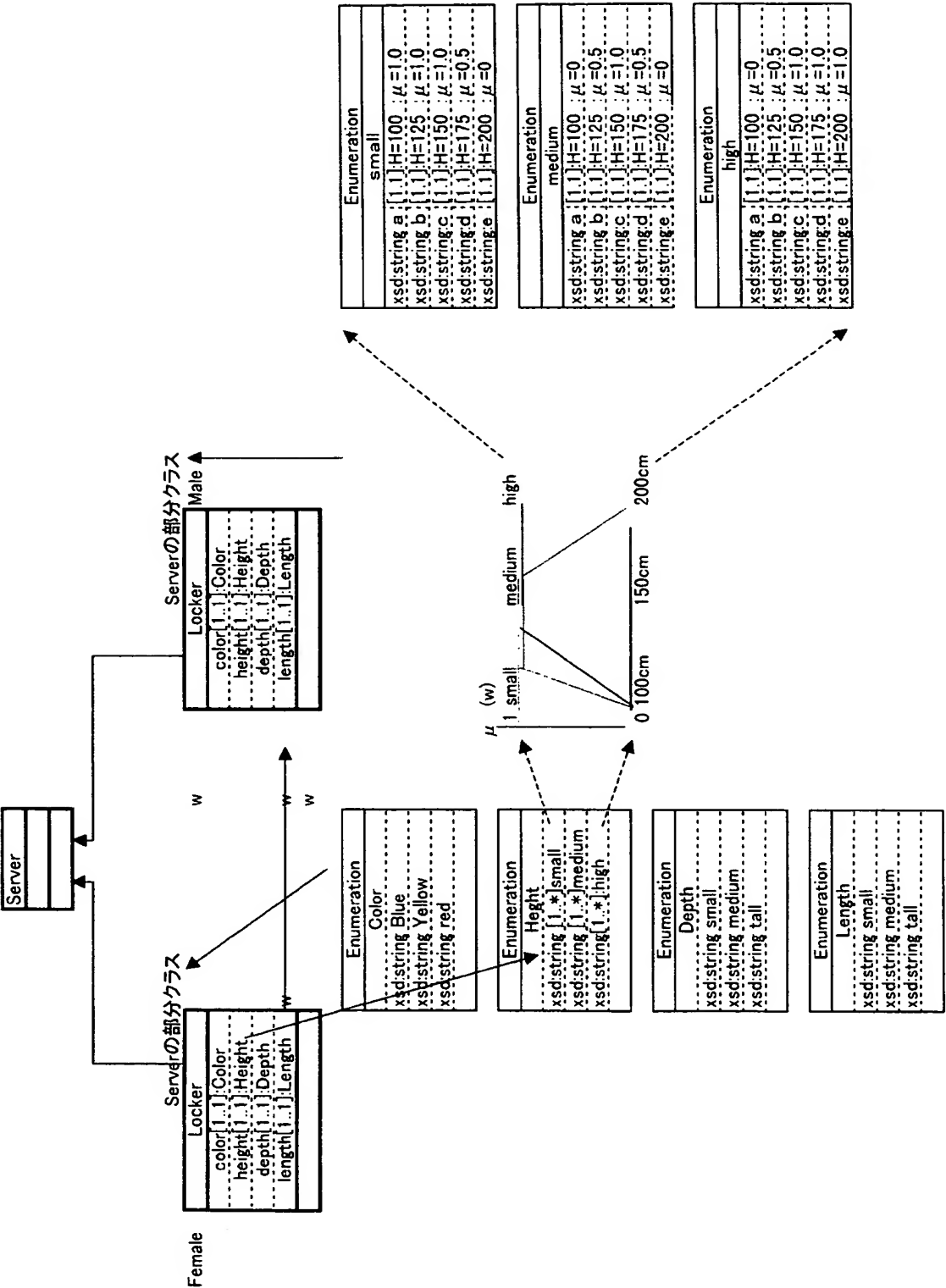
【図 11】

知識構造とクラスの関係を示す図(その3)



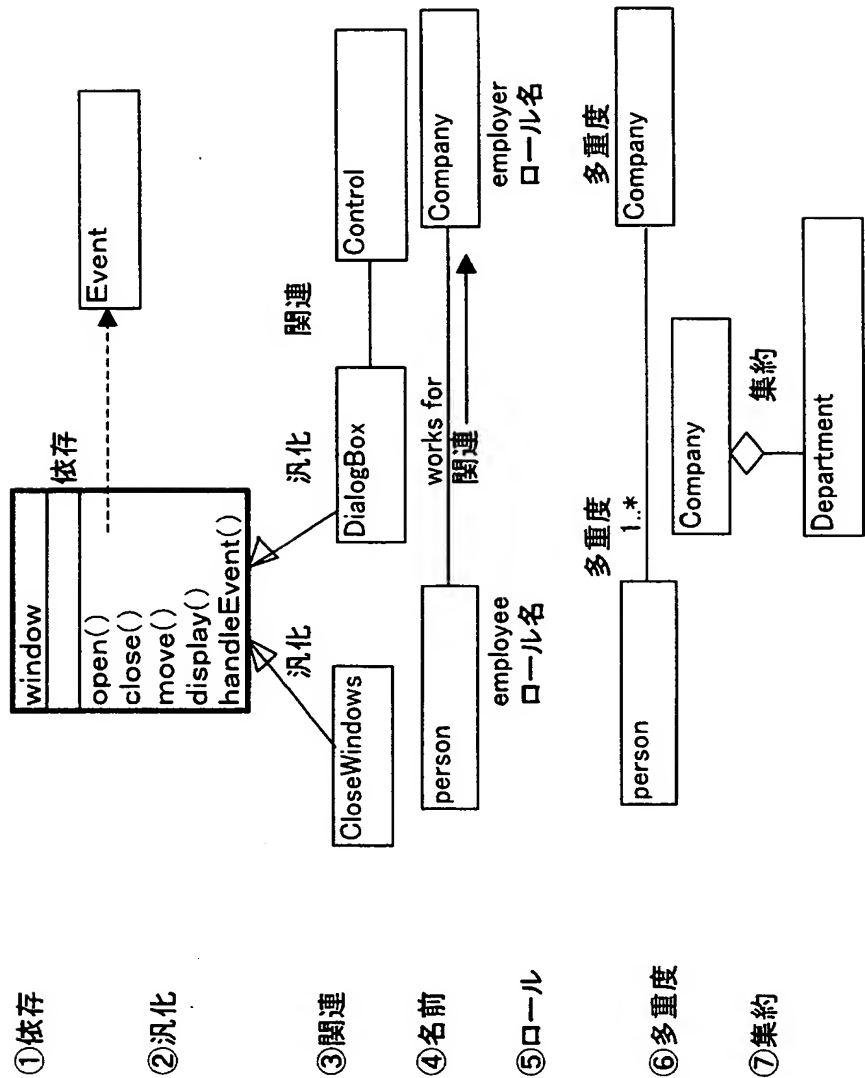
【図 12】

知識構造とクラスの関係を示明する図(その4)



【図 13】

従来技術を説明する図



【書類名】 要約書

【要約】

【課題】 本発明の課題は、設計者、利用者の負担を減らしつつ、正確に知識設計することの出来る知識処理システムを提供する。

【解決手段】 設計側では、構造化された知識構造を持つクラス群に、全体を統括するコードネームを持つスーパークラスを生成する。そして、スーパークラスを参照することによって、クラス群間の関係を多値論理で推論し、関連付ける。また、可能な場合には、多値論理による推論で新しいクラスを生成する。利用側では、設計側で生成された知識構造を示すクラス群とそれらの関係の他に、独自に多値論理による推論を行い、クラス群の間関係を再構築し、クラスとクラス間の関係として記述された知識構造を利用する。

【選択図】 図 1

特願 2 0 0 3 - 0 7 9 4 6 6

出 願 人 履 歴 情 報

識別番号

[0 0 0 0 0 5 2 2 3]

1. 変更年月日

1 9 9 6 年 3 月 2 6 日

[変更理由]

住所変更

住 所

神奈川県川崎市中原区上小田中 4 丁目 1 番 1 号

氏 名

富士通株式会社